

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of)	
Guido et al.)	Patent Pending
)	
Serial No.: 10/814,551)	Examiner: Henry Orr
)	
Filed: March 31, 2004)	Group Art Unit: 2176
)	
For: Affinity Group Window Management)	Confirmation No.: 7434
System and Method)	
)	
Attorney's Docket No: 4541-016)	

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

CERTIFICATE OF MAILING OR TRANSMISSION [37 CFR 1.8(a)]

I hereby certify that this correspondence is being:

- ☐ deposited with the United States Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to: Mail Stop Appeal Brief Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.
- ☐ transmitted by facsimile on the date shown below to the United States Patent and Trademark Office at (703) 273-8300.

Date

This correspondence is being:

- ☒ electronically submitted via EFS-Web

REPLY BRIEF

The prior art does not disclose grouping different and independent applications.

The independent claim recites grouping GUI windows running different and independent applications, and simultaneously altering the z-order of all windows in the group. Appellants have urged that applications are *different* if they are not the same application, and are *independent* if they are not logically or functionally related to each other (*e.g.*, as parent/child). The combination of Ashe and Diedrichsen fails to teach or suggest grouping GUI windows for

simultaneous z-order manipulation, where the windows run applications that are different and independent from each other.

As one example of grouping GUI windows running different and independent applications, the specification describes, at ¶ 0014, and with reference to Figure 2, a window 40 associated with a word processor application, a window 38 associated with an e-mail client, and a window 36 associated with a web browser. The user may define an affinity group comprising the windows 36, 38, 40. Thereafter, whenever one of the windows 36, 38, 40 is selected, all three windows 36, 38, 40 rise to the top of the GUI desktop (*i.e.*, they overlie, or obscure, all other windows). The applications are *different* in that they are not the same application, and are *independent* in that they are not logically or functionally related *to each other*.

On page 18 of the Answer, the Examiner quotes from the Specification, ¶ 0014, “In this case, the Internet browser in window 36, the e-mail client in window 38, and the word processor in window 40 are all related *to the same task or operation* – namely, writing a report.” Those of skill in the art – indeed, anyone skilled in using a personal computer – knows that a browser, e-mail client, and word processor are applications that are not related to each other. Each performs a separate task, stores data in a separate format, etc. At any given time, the applications may be related *by the task to which the user sets them*. Indeed, this is the only reason a user may wish to group them for z-order manipulation.

By analogy, an airplane pilot may consider a map, compass, and stopwatch “related” when applying the three items to the task of navigation. He may thus wish to store and access the three items together, much like simultaneously popping three GUI windows to the top of the desktop. However, the items are only “related” to each other by virtue of the task to which they are applied. The stopwatch, for example, is otherwise completely unrelated to either a map or a

compass. Similarly, the application programs of a web browser, e-mail client, and word processor are unrelated to each other, although a user may consider them temporarily “related” as being employed in the same task. Thus, there is no conflict between Appellants’ proffered definition of independent as unrelated to each other, and the disclosure in Appellants’ Specification of independent applications being “related to the same task or operation.”

The window grouping disclosed in Diedrichsen is between windows running applications that are inherently and inextricably related as parent/child. Indeed, Diedrichsen relies on that relationship to be able to maintain the grouping (traversing the parent/child tree to locate each window in the group). The applications are related, regardless of the end use to which the user applies them. Thus, Diedrichsen fails to teach or suggest simultaneous manipulation of the z-order of windows running different and independent applications, where independent means not related to each other.

The § 103 rejections are not supported by the prior art references cited.

In the Appeal Brief, Appellants analyzed what both Ashe and Diedrichsen do and do not disclose, concluding that neither reference teaches or suggests grouping windows for z-order manipulation where the windows are associated with different and independent applications. Ashe does not teach this. Diedrichsen does not teach this. Unable to refute these showings, the Examiner argues that the rejections do not in fact rely on the Examiner’s citations to the prior art references, but somehow magically find support in their combination.

In response to Appellants’ showing that Ashe discloses grouping windows associated with the same application, the Examiner states, on p. 19,

Examiner notes to Appellant that the Examiner does not rely solely on Ashe to teach or suggest an affinity group of GUI windows and manipulating the z-

order of the group, where the windows are associated with at least two different applications. In fact, Examiner submits that such limitation would have been obvious over Ashe in view of Diedrichsen.

In response to Appellants' showing that Ashe does not disclose grouping windows associated with different application, the Examiner states, on p. 20,

As explained above, Examiner does not rely solely on Ashe to teach or suggest grouping windows associated with different and independent applications. Examiner relies on Ashe in view of Diedrichsen to teach such features as explained above.

In response to Appellants' showing that Diedrichsen discloses grouping windows related as parent/child, the Examiner states, on p. 22,

Examiner notes to Appellant that the Examiner does not rely solely on Diedrichsen to teach or suggest grouping windows, for simultaneous z-order manipulation, that are associated with independent applications. In fact, Examiner submits that such limitation would have been obvious over Ashe in view of Diedrichsen.

Finally, in response to Appellants' showing that Diedrichsen does not disclose grouping windows associated with independent application, the Examiner states, on p. 24,

As stated above, Examiner notes to Appellant that the Examiner does not rely solely on Diedrichsen to teach or suggest grouping windows associated with different and independent applications for z-order manipulation. Examiner submits that such limitation would have been obvious over Ashe in view of Diedrichsen as explained above.

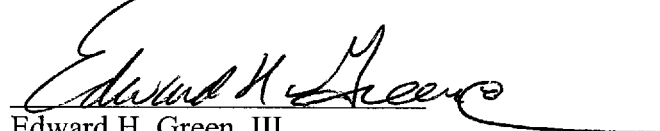
"Ashe in view of Diedrichsen" cannot – and does not – teach or suggest what neither reference discloses. Ashe teaches only grouping windows running parts of the *same* application. Diedrichsen teaches only grouping windows running applications *related to each other* as parent/child. Neither Ashe nor Diedrichsen – separately or in combination – fairly teach or suggest grouping windows for z-order manipulation where the windows are associated with *different and independent* applications, as claimed.

Conclusion

For the reasons discussed in the Appeal Brief and herein, the § 103 rejections of claims 1, 14, 19, and 25, and all claims depending therefrom, must be reversed.

Respectfully submitted,

COATS & BENNETT, P.L.L.C.

A handwritten signature in black ink, appearing to read "Edward H. Green, III", is written over a horizontal line.

Edward H. Green, III
Registration No.: 42,604

Dated: November 18, 2008

1400 Crescent Green, Suite 300
Cary, NC 27518
Telephone: (919) 854-1844
Facsimile: (919) 854-2084